

컴퓨터 물리 해석 1차 보고서 | Spontaneous Synchronization

2503 김도윤

1. 목적

이와 같이 여러 개의 메트로놈이 하나의 받침대 위에 있는 상황에서 시간이 지나면 서로 동기화하게 된다. 이 현상이 신기했고, 동기화에 어떤 조건이 있는지 알고 싶었다. 또한 위상이 수렴하는 양상이 잘 상상 안되었기에 이에 대해 알아보고 싶었으며, 동기화에 제일 중요한 요소가 완벽한 단진동이 아닌 것, 공기저항 등 중에서 어떤 것인지 알아보고 싶었다.

2. 물리적 의미

Spontaneous synchronization은 서로 약하게 연결된 다수의 진동자들이 공통된 매개체를 통해 상호작용하면서, 외부의 강제 없이도 스스로 위상이나 주기를 일치시키는 현상이다.

이를 자유도(degree of freedom)의 관점에서 설명하면, 독립적이던 진동자들이 매개체를 통해 하나의 연동된 자유도를 공유하게 되면서, 전체 시스템이 에너지적으로 안정된 동기화 상태로 수렴하는 것이다.

수학적으로는 Kuramoto 모델이 대표적인 예로, 진동자의 위상 θ_i 가 다음과 같이 기술된다:

$$\frac{d\theta_i}{dt} = \omega_i + \frac{K}{N} \sum_{j=1}^N \sin(\theta_j - \theta_i)$$

여기서 ω_i 는 자연 진동수, K 는 상호작용 강도이며, K 가 임계값 이상이 되면 위상 차이가 점점 줄어들며 동기화가 자발적으로 이루어진다.

3. 물리적 유도

물리량	기호	기본값
받침대의 위치	X	
각 진자의 각도	θ_i	
진자의 길이	r	1
진자의 질량	m	1
받침대의 질량	M	100
저항력 (\propto 각속도)	b	0.1
기타 받침대가 진자에게 주는 힘	F_i	

각 진자 끝의 속도는

$$(\dot{X} + r\dot{\theta}_i \cos \theta_i, -r\dot{\theta}_i \sin \theta_i)$$

로 나타내지며, 이의 속도²은

$$\dot{X}^2 + 2r \cos \theta_i \dot{X} \dot{\theta}_i + r^2 \dot{\theta}_i^2$$

이다. 따라서 전체 운동에너지는

$$T = \sum_{i=1}^n \frac{1}{2} m (\dot{X}^2 + 2r \cos \theta_i \dot{X} \dot{\theta}_i + r^2 \dot{\theta}_i^2) + \frac{1}{2} M \dot{X}^2$$

라는 것을 알 수 있다. 또한, 전체 포텐셜 에너지는

$$V = - \sum_{i=1}^n mgr \cos \theta_i$$

이다.

이때 라그랑지안을 보면:

$$\begin{aligned} L &= \sum_{i=1}^n \frac{1}{2} m \dot{X}^2 + \sum_{i=1}^n mr \cos \theta_i \dot{X} \dot{\theta}_i + \sum_{i=1}^n \frac{1}{2} mr^2 \dot{\theta}_i^2 + \sum_{i=1}^n mgr \cos \theta_i + \frac{1}{2} M \dot{X}^2 \\ \frac{\delta L}{\delta \dot{\theta}_i} &= mr \cos \theta_i \dot{X} + mr^2 \dot{\theta}_i \\ \frac{d}{dt} \frac{\delta L}{\delta \dot{\theta}_i} &= mr (-\sin \theta_i \dot{\theta}_i) \dot{X} + mr \cos \theta_i \ddot{X} + mr^2 \ddot{\theta}_i \\ \frac{\delta L}{\delta \theta_i} &= mr (-\sin \theta_i) \dot{X} \dot{\theta}_i + mgr (-\sin \theta_i) \\ \therefore mr \cos \theta_i \ddot{X} + mr^2 \ddot{\theta}_i &= -mgr \sin \theta_i, \\ mr \ddot{\theta}_i &= -mg \sin \theta_i - m \cos \theta_i \ddot{X} \\ \frac{\delta L}{\delta \dot{X}} &= (nm + M) \dot{X} + mr \sum_{i=1}^n \cos \theta_i \dot{\theta}_i \\ \frac{d}{dt} \frac{\delta L}{\delta \dot{X}} &= (nm + M) \ddot{X} + mr \sum_{i=1}^n (-\sin \theta_i \dot{\theta}_i) \dot{\theta}_i + mr \sum_{i=1}^n \cos \theta_i \ddot{\theta}_i \\ \frac{\delta L}{\delta X} &= 0 \\ \therefore (nm + M) \ddot{X} + mr \sum_{i=1}^n (-\sin \theta_i \dot{\theta}_i) \dot{\theta}_i + mr \sum_{i=1}^n \cos \theta_i \ddot{\theta}_i &= 0 \end{aligned} \quad (1)$$

저항력 $b\dot{\theta}_i$, 추가 힘 $F_i \cos \theta_i$ 를 추가하면:

$$\begin{aligned} mr \ddot{\theta}_i &= -mg \sin \theta_i - m \cos \theta_i \ddot{X} - bmr \dot{\theta}_i + F_i \cos \theta_i \\ mr \cos \theta_i \ddot{\theta}_i &= -mg \sin \theta_i \cos \theta_i - m \cos^2 \theta_i \ddot{X} - bmr \cos \theta_i \dot{\theta}_i + F_i \cos^2 \theta_i \end{aligned} \quad (1')$$

이를 (2)에 대입하면:

$$\begin{aligned} (nm + M) \ddot{X} - mr \sum_{i=1}^n \sin \theta_i \dot{\theta}_i^2 \\ - mg \sum_{i=1}^n \sin \theta_i \cos \theta_i \\ - m \ddot{X} \sum_{i=1}^n \cos^2 \theta_i \\ - \sum_{i=1}^n bmr \cos \theta_i \dot{\theta}_i \\ + \sum_{i=1}^n F_i \cos^2 \theta_i &= 0 \end{aligned}$$

따라서:

$$\ddot{X} = \frac{mr \sum_{i=1}^n \sin \theta_i \dot{\theta}_i^2 + mg \sum_{i=1}^n \sin \theta_i \cos \theta_i + \sum_{i=1}^n bmr \cos \theta_i \dot{\theta}_i - \sum_{i=1}^n F_i \cos^2 \theta_i}{(nm + M) - m \sum_{i=1}^n \cos^2 \theta_i}$$

또한 (1')에서,

$$\ddot{\theta}_i = \frac{-mg \sin \theta_i - m \cos \theta_i \ddot{X} - bmr\dot{\theta}_i + F_i \cos \theta_i}{mr}$$

이는 이와 같이 구현한다:

```
...

def acceleration(self, x: Data, v: Data, f: np.ndarray) → Data:

    dividend = self.m * np.sum(self.rs * np.sin(x.x) * v.x ** 2) \
        + self.m * g * np.sum(np.sin(x.x) * np.cos(x.x)) \
        + self.b * self.m * np.sum(self.rs * np.cos(x.x) * v.x) \
        - np.sum(f * np.cos(x.x) ** 2)

    divisor = (self.n * self.m + self.M) - self.m * np.sum(np.cos(x.x))

    A = dividend / divisor

    a = -g * np.sin(x.x) / self.rs \
        - np.cos(x.x) * A / self.rs \
        - self.b * v.x \
        + f * np.cos(x.x) / self.m / self.rs

    ...
```

운동량 보존, ($b = 0$ 일때) 에너지 보존 등으로 위 구현의 정확성을 확인할 수 있었다.

메트로놈은 단진자 뿐만 아니라 중심을 지날 때마다 진자에게 추가적인 힘을 주는 태엽 장치가 있다. 이 시뮬레이션에선 중심에서 멀어지는 쪽으로 `angle_threshold` (기본값 0.1)를 넘었을 때 `impulse` (기본값 0.1)의 충격량을 주는 방법으로 이를 구현했다:

4. 구현

$$\vec{X} = [x_1, x_2, \dots, x_n, X]$$

와 같이 변수들을 묶어주는 `Data` 클래스를 구현하고, 이에 대해 `__add__`, `__mul__` 등의 사칙연산을 구현해 하나의 값으로 다룰 수 있도록 하였다.

시뮬레이션의 state를 관리하는 `Simulation` 클래스와 이를 `vpython`으로 시각화하는 `Visualizer` 클래스를 만듦으로써 둘의 코드를 분리해 데이터 수집 시 불필요한 시각화를 하지 않아도 되도록 하였고, 새로운 클래스를 생성하는 방법으로 매개변수가 다른 새로운 시뮬레이션을 간단히 만들 수 있도록 하였다.

진자에게 초기 충격량을 줄 수 있는 버튼과 시뮬레이션 속도를 바꿀 수 있는 슬라이더를 추가해 코드를 만지지 않고 활용할 수 있도록 하였고, 매개변수 섹션과 <Restart> 버튼을 따로 만들어 시뮬레이션의 여러 개의 요소를 바꿔가면서 실행할 수 있도록 하였다.

x-t 그래프와 최대 진폭, 그리고 위상 ($= \arctan 2(x, v)$) 그래프를 제공함으로써 시뮬레이션의 양상을 확인할 수 있도록 하였다.

5. AI 활용

1차 시도에는 AI를 중심으로 코드 작성을 하며 `vpython`이라는 새로운 라이브러리를 씬에 불과하고 프로토타입을 빨리 만들 수 있었다. 다만 이 프로토타입은 코드 길이가 늘어나자 프롬프팅을 사용한 관리는 어려워졌고, 직접 개발 시에는 리팩터링이 많이 필요해 폐기하였다. 이후에는 직접 `vpython` 문서를 참조하고 클래스를 설계하여 개발을 진행했다.

오히려 물리 부분에서 도움을 받았다. 처음 라그랑지안 계산 중 미분이 틀리게 되어 물리적으로 설명이 되지 않는 결과가 나왔는데, 수식을 검증받으며 고칠 수 있었다. 또한 이후 외부 힘을 추가하는 과정에서 운동량 보존이 정확히 되지 않는 점을 발견했는데, 또한 시와 상호 교섭한 결과 틀린 항을 발견할 수 있었다.

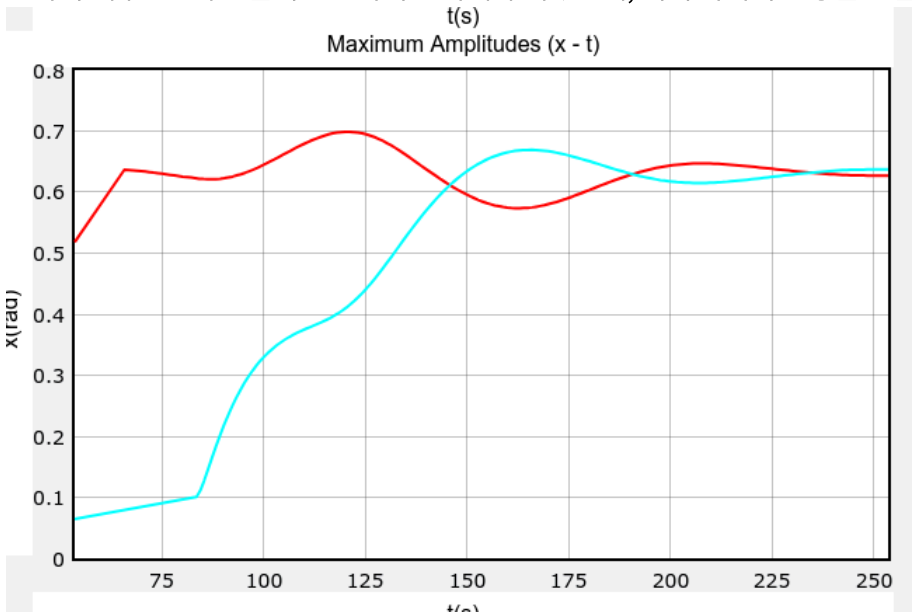
6. 결과

시간이 지날수록 진자의 위상이 모두 같아지는 것을 관찰할 수 있었다. 정성적으로 관찰한 바로는:

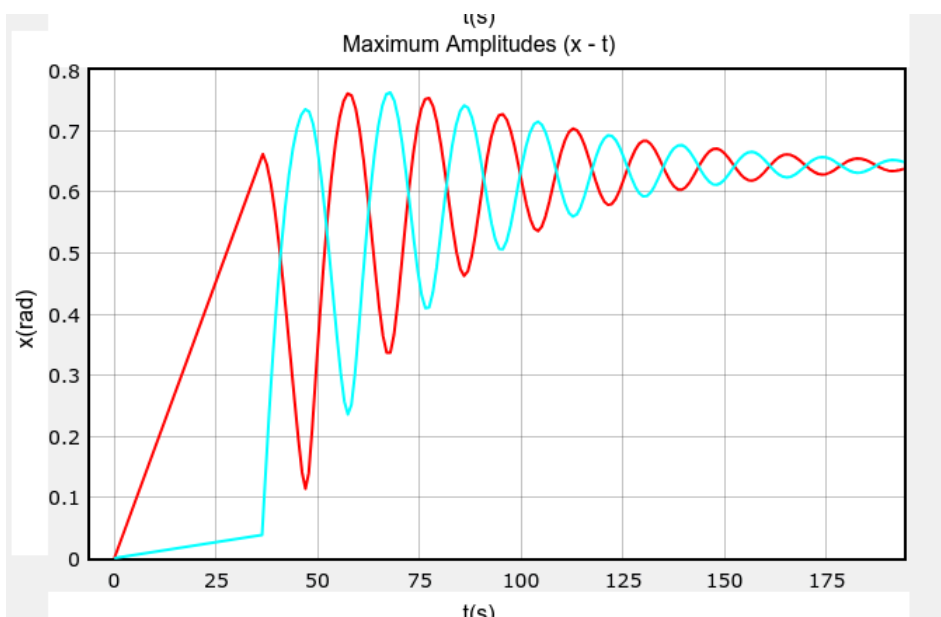
- b 가 클수록, M 이 작을수록 위상 동기화가 더 빠르게 되었다.
- 모두 같은 위상으로 진동하는 경우만 관찰할 수 있었고, 위상이 π 만큼 다른 상태로 진동하는 경우는 일시적으로 존재 하긴 했지만 불안정해 모두 같은 위상으로 돌아갔다.
- 주기는 \sqrt{r} 에 대략적으로 반비례했다.
- $b = 0$ 인 경우에는 동기화가 일어나지 않으며, 완벽한 단진동으로 근사 ($\sin x = x, \cos x = 1$)하는 것은 영향을 주지 않는다.

(1) M 에 따른 진자의 진폭 양상

진자의 개수를 2개로 줄이고 진자의 진폭에 주목했는데, 여러 가지의 현상을 관찰할 수 있었다.



($M = 100$ 일 때의 진자의 진폭 양상)

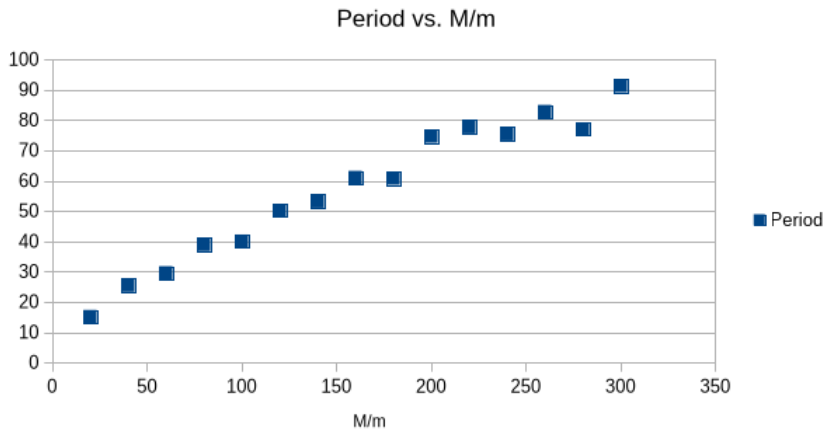


(M = 10일 때의 진자의 진폭 양상)

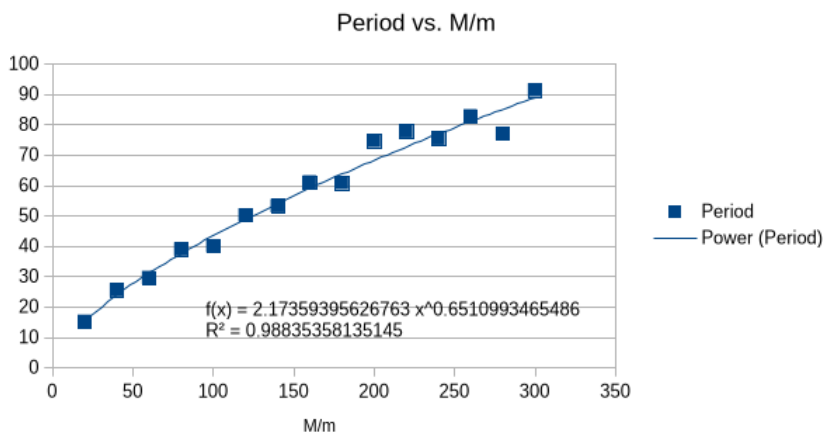
진폭은 어떤 점에 대해서 감쇠진동하는 양상을 보였고, 이 양상의 주기를 구해 보았다.

- 일단 `amplitudes.py` 에 최대진폭을 기록하는 함수를 구현했다.
- M이 20과 300 사이일 때의 (시간, 최대진폭)을 `output/{M}.csv` 파일로 저장하는 `record_amplitudes.py` 를 작성하였다.
- 두 진자의 진폭이 서로 교차하는 지점을 구하고, 이에 대해 선형 회귀해 주기를 구하는 `record_intersections.py` 를 작성하였다. `periods.csv` 에서 이를 볼 수 있다.

결과값들은 다음과 같다:



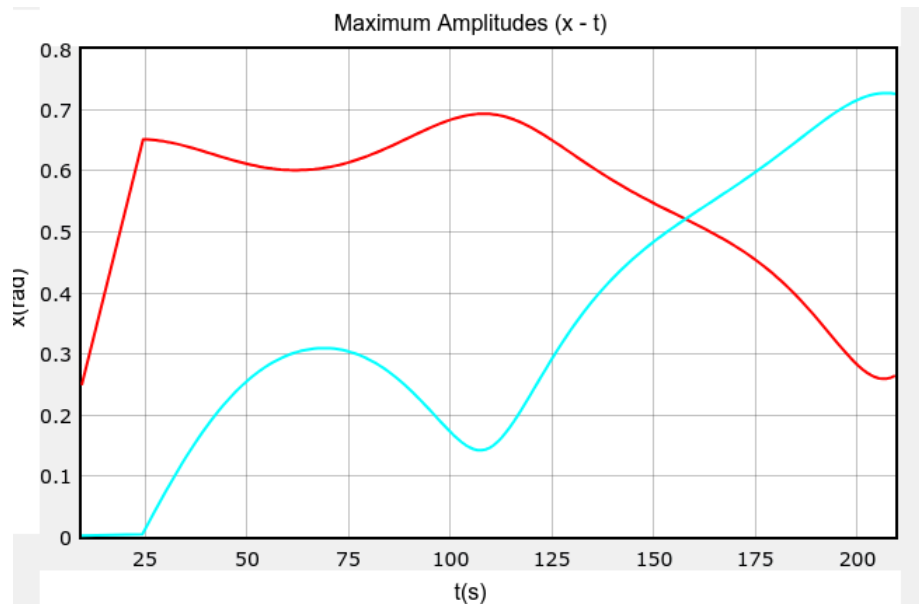
추세선 분석 결과, 비례한다고 가정했을 때와 제곱근과 비례한다고 가정했을 때 둘 다 0.98 정도의 R^2 값이 나왔고, Power함수로 가정했을 때:



이와 같은 결과가 나왔다. 다만 지수값 0.65에 대해서는 추후 탐구가 필요할 듯 하다.

(2) b (저항값)에 따른 진자의 진폭 양상

b는 진자의 움직임을 받침대에 전달하는 역할을 하므로, b가 작으면 동기화하는 시간도 느려지는 것을 확인할 수 있었다.

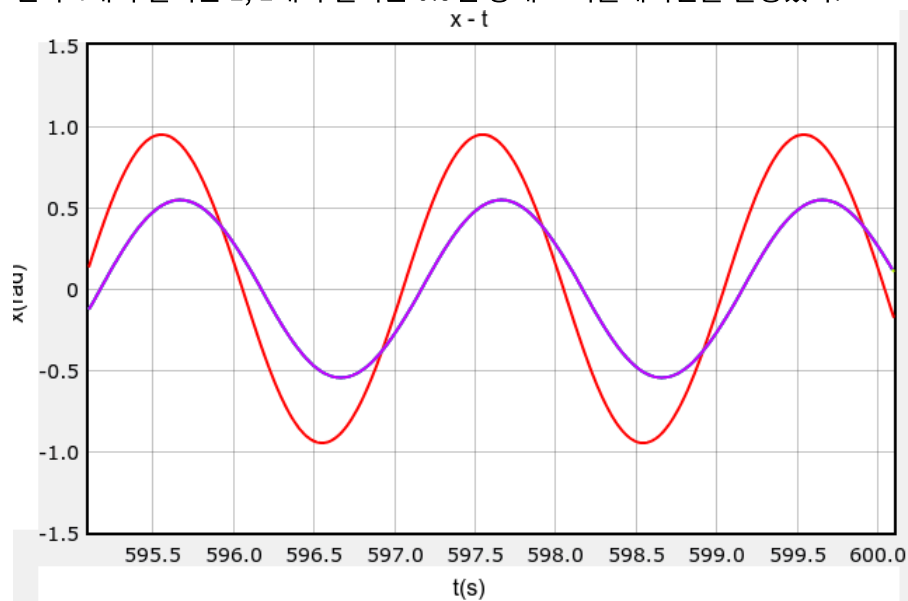


($b = 0.01$ 일 때의 진자의 진폭 양상)

저항값이 작은 경우에는 위와 같이 진폭 차이가 줄어드는 구간도 있고 커지는 구간도 있는데, 위상 그래프와 대조해서 보면 이것은 위상 차이가 π 보다 큰 상황에서 에너지가 반대로 전달되는 경우인 것을 확인할 수 있다.

(3) 진자의 고유 진동수가 다른 경우의 진자의 진폭 및 위상 양상

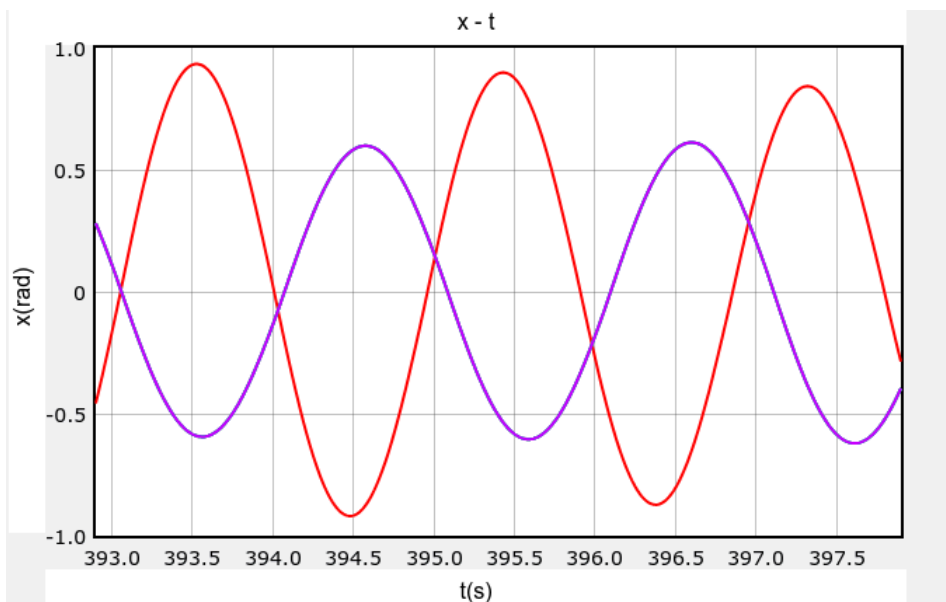
진자 4개의 길이는 1, 1개의 길이는 0.9인 상태로 시뮬레이션을 실행했다.



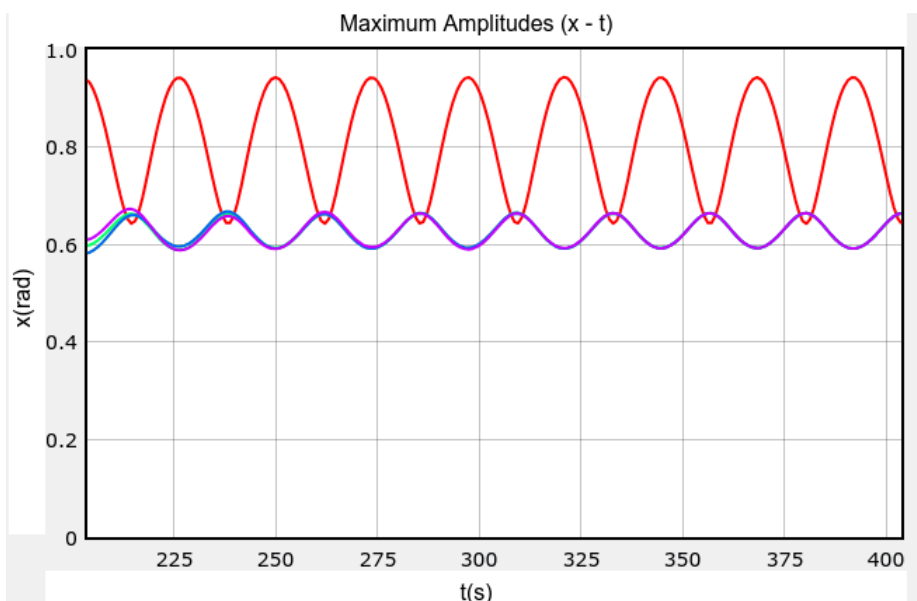
(빨강: 0.9)

이와 같이 길이가 짧은 (주기가 짧은) 의 위상이 약간 앞서고, 진폭이 더 큰 것을 관찰할 수 있었다.

1개의 길이를 0.8로 더 줄이면:



이와 같이 더 이상 동기화 되지 않으며,



이와 같이 위상 차이가 변동하면서 각 진자의 진폭도 같이 변동하는 것을 관찰할 수 있었다. 이분탐색으로 반복 실험한 결과 이 임계값은 0.8625와 0.875 사이임을 알 수 있었다.

7. 보완할 점

각 진자가 받침대에 어느 정도의 힘을 가하고 있고, 이들이 다른 진자의 움직임을 돕는지 반하는지에 대해 시각화 및 분석을 할 계획이었지만, 시간이 부족해 할 수 없었다. 각 진자의 길이 (즉, 고유 진동수)가 다른 경우에 대해서도 시뮬레이션을 하였지만, 이를 GUI를 사용하여 조절할 수 있게 하진 못했다.

또한, 각 진자의 길이가 다른 경우에는 b 와 M 과 같은 다른 변수에 의해서도 임계값이 달라지는데, 이를 자동으로 측정하기 위해선 동기화했는지에 대한 판단 기준이 필요하며, 훨씬 성능이 좋은 시뮬레이션이 필요하기 때문에 하지 못했다.

8. 협업 및 소통

소프트 바디 시뮬레이션 방식으로 천을 시뮬레이션하는 친구가 있었는데, 전에 소프트 바디 시뮬레이션을 경험해본 자로서 시뮬레이션의 어떤 부분에 허점이 있는지(각 노드와만 충돌을 하고, 면과는 충돌을 하지 않는 것)를 알려주었다. 또한 이때 정확한 물리적 계산은 어려우니 근사를 어떻게 할 수 있는지 설명해 주었다.

vpython은 파이썬 버전이 매우 낮고, 파일 여러 개를 쓸 수 없기 때문에 로컬로 프로젝트를 개발하자는 조언을 여러 번 하였다.

이 주제를 잡기 전 Physically Based Dynamics에 대해 조사하고 구현했는데, 이때 이것이 전통적인 시뮬레이션과 어떻게 다른지 설명하면서 다른 시뮬레이션에도 사용될 수 있는 새로운 접근 방향을 제시했다.

쉬운 관리를 위해 프로젝트를 깃허브에 업로드했는데, 실행이 어렵다는 피드백이 있어 README에 실행 관련 설명을 추가했다.

단진자 뿐만 아니라 용수철 진자에서도 이와 같은 현상이 일어나는지 보고 싶다는 피드백이 있었는데, 비록 코드를 따로 만들진 않았지만 기존 코드를 수정하여 운동을 확인하였다.

또한 진자의 움직임을 더 많이 가시화하면 좋겠다는 피드백이 있었는데, 최대진폭 그래프와 위상 그래프를 추가하며 이를 반영했다.

9. 부록: 각 진자의 길이가 다른 경우의 물리적 유도

$$T = \sum_{i=1}^n \frac{1}{2} m (\dot{X}^2 + 2r_i \cos \theta_i \dot{X} \dot{\theta}_i + r_i^2 \dot{\theta}_i^2) + \frac{1}{2} M \dot{X}^2$$

$$V = - \sum_{i=1}^n mgr_i \cos \theta_i$$

$$L = \sum_{i=1}^n \frac{1}{2} m \dot{X}^2 + \sum_{i=1}^n mr_i \cos \theta_i \dot{X} \dot{\theta}_i + \sum_{i=1}^n \frac{1}{2} mr_i^2 \dot{\theta}_i^2 + \sum_{i=1}^n mgr_i \cos \theta_i + \frac{1}{2} M \dot{X}^2$$

$$\frac{\delta L}{\delta \dot{\theta}_i} = mr_i \cos \theta_i \dot{X} + mr_i^2 \dot{\theta}_i$$

$$\frac{d}{dt} \frac{\delta L}{\delta \dot{\theta}_i} = mr_i (-\sin \theta_i \dot{\theta}_i) \dot{X} + mr_i \cos \theta_i \ddot{X} + mr_i^2 \ddot{\theta}_i$$

$$\frac{\delta L}{\delta \theta_i} = mr_i (-\sin \theta_i) \dot{X} \dot{\theta}_i + mgr_i (-\sin \theta_i)$$

$$\therefore mr_i \cos \theta_i \ddot{X} + mr_i^2 \ddot{\theta}_i = -mgr_i \sin \theta_i,$$

$$mr_i \ddot{\theta}_i = -mg \sin \theta_i - m \cos \theta_i \ddot{X}$$

$$\frac{\delta L}{\delta \dot{X}} = (nm + M) \dot{X} + m \sum_{i=1}^n r_i \cos \theta_i \dot{\theta}_i$$

$$\frac{d}{dt} \frac{\delta L}{\delta \dot{X}} = (nm + M) \ddot{X} + m \sum_{i=1}^n r_i (-\sin \theta_i \dot{\theta}_i) \dot{\theta}_i + m \sum_{i=1}^n r_i \cos \theta_i \ddot{\theta}_i$$

$$\frac{\delta L}{\delta X} = 0$$

$$\therefore (nm + M) \ddot{X} + m \sum_{i=1}^n r_i (-\sin \theta_i \dot{\theta}_i) \dot{\theta}_i + m \sum_{i=1}^n r_i \cos \theta_i \ddot{\theta}_i = 0$$

$$mr_i \ddot{\theta}_i = -mg \sin \theta_i - m \cos \theta_i \ddot{X} - bmr_i \dot{\theta}_i + F_i \cos \theta_i$$

$$mr_i \cos \theta_i \ddot{\theta}_i = -mg \sin \theta_i \cos \theta_i - m \cos^2 \theta_i \ddot{X} - bmr_i \cos \theta_i \dot{\theta}_i + F_i \cos^2 \theta_i$$

$$(nm + M) \ddot{X} - m \sum_{i=1}^n r_i \sin \theta_i \dot{\theta}_i^2 - mg \sum_{i=1}^n \sin \theta_i \cos \theta_i - m \ddot{X} \sum_{i=1}^n \cos^2 \theta_i - \sum_{i=1}^n bmr_i \cos \theta_i \dot{\theta}_i + \sum_{i=1}^n F_i \cos^2 \theta_i = 0$$

$$\ddot{X} = \frac{m \sum_{i=1}^n r_i \sin \theta_i \dot{\theta}_i^2 + mg \sum_{i=1}^n \sin \theta_i \cos \theta_i + \sum_{i=1}^n bmr_i \cos \theta_i \dot{\theta}_i - \sum_{i=1}^n F_i \cos^2 \theta_i}{(nm + M) - m \sum_{i=1}^n \cos^2 \theta_i}$$